



2023. 07. 27

금오공과대학교 SSL 세미나

# 쉬운 길 돌아가기 전문가

박준수

[취준생/만 23세]

## 01

### 서류

- 산업체 현장 실습
  - ICT 인턴십
- 

## 02

### 면접

- 탈탈탈
  - 비대면 화상 면접
  - 대면 면접
- 

## 03

### 처음부터 다시

- 정보처리기사 / 토익
- 코딩테스트 / 프로젝트

# 01

## 서류 전형

- 현장 실습
- ICT 인턴십

# 이번 결과..

## 백엔드 개발 직무

지역	프로젝트명	매칭상태	면접일자	기업선택
서울 서초구	[프로젝트명]	서류전형 불합격		
서울 구로구		면접전형 불합격	2023-07-24	
서울 영등포구		서류전형 불합격		

NestJS 사용

Java 사용

Javascript 사용

# 이력서 & 자기소개서

## 이력서 박준수

인적사항	성명	박준수
	생년월일(나이)	1999년 08월 20일 ( 만 23세 )
	성별	남
	주소	
	휴대전화	
	E-mail	
학력사항	대학명	금오공과대학교
	학과명	컴퓨터공학과
	학년	4학년
	평균학점(전학년)	3.85점 / 4.5점
	전공평균학점(전학년)	3.90점 / 4.5점
	상태	재학
관심 직무(기술)분야	관심 분야 1	웹 백엔드 (서버 개발)
	관심 분야 2	웹 백엔드 (배포, 클라우드)
	관심 분야 3	웹 프론트엔드
기타 특이사항	- 없음	
포트폴리오	- Github <a href="https://github.com/BearWithpy">https://github.com/BearWithpy</a>	
	- 포트폴리오 <a href="https://comet-universe-415.notion.site/44dc7815d14341b890d2cfa40dd12026?vs=4">https://comet-universe-415.notion.site/44dc7815d14341b890d2cfa40dd12026?vs=4</a>	

위 기재한 사항은 사실임을 확인합니다.

2023 년 7 월 7 일

성 명 : 박 준 수 

ICT 학원연계 프로젝트 인턴십

## 전공이수학점내역 박준수

구분	교과목명	학점	이수학점	이수학년
전공필수	컴퓨터공학입문	Pass	1	1
전공필수	창의공학설계입문	B+(3.5)	3	1
전공필수	기초프로그래밍	B+(3.5)	3	1
전공필수	디지털공학	A+(4.5)	3	1
전공필수	디지털공학실험	B+(3.5)	1	2
전공필수	C프로그래밍	B(3.0)	3	1
전공필수	C프로그래밍응용	B(3.0)	2	1
전공선택	웹프로그래밍	B+(3.5)	3	2
전공선택	전자시스템	B(3.0)	3	2
전공필수	자바프로그래밍	A(4.0)	3	2
전공필수	자료구조	A(4.0)	3	2
전공필수	자료구조응용및실습	B(3.0)	1	2
전공필수	컴퓨터구조	A+(4.5)	3	2
전공선택	리눅스활용및프로그래밍	B+(3.5)	3	2
전공필수	소프트웨어분석및설계	A+(4.5)	3	2
전공선택	GUI프로그래밍	A+(4.5)	3	2
전공선택	IoT기초설계	B+(3.5)	3	2
전공필수	운영체제	B+(3.5)	3	3
전공필수	컴퓨터네트워크	A+(4.5)	3	3
전공선택	데이터베이스	A+(4.5)	3	3
전공선택	프로그래밍언어개념	A(4.0)	3	3
전공선택	영상처리	A(4.0)	3	3
전공선택	소프트웨어공학	B+(3.5)	3	3
전공선택	임베디드시스템	A+(4.5)	3	3
전공필수	임베디드시스템실습	A+(4.5)	1	3
전공선택	스마트앱프로그래밍	A(4.0)	3	3
전공선택	컴퓨터그래픽스	A(4.0)	3	3
전공필수	오픈소스프로젝트	A+(4.5)	3	3
전공학점 소개			75	
평균		3.90	75	

ICT 학원연계 프로젝트 인턴십

## 자기소개서 박준수

학교에서 실시하는 프로그램을 통해 기업과 협업하여 프로젝트를 진행할 수 있는 경험을 하였습니다. 기업에서 제안한 프로젝트는 임원진들의 경영 의사결정을 지원하는 EIS를 개발하는 것이었습니다. 프로젝트 크기가 작아 스프링을 사용하지 않고 jsp와 was로 Tomcat을 두고 프로젝트를 진행하였습니다. 저는 기업에서 제공해주는 Windows 서버 위에 Tomcat을 올려 배포하는 일과 jdbc driver를 이용하여 사용할 mssql 데이터베이스 연결하는 부분을 맡았습니다. 그런데 프로젝트를 진행하면서 데이터베이스 연결하다가 문제를 만나게 되었습니다. jdbc 드라이버로 데이터베이스에 접근 테스트를 해보았는데 보안 연결을 설정할 수 없다는 에러 내용을 보았습니다. 드라이버 연결 코드의 문제가 없는 것 같아 에러의 내용인 보안 연결에 대하여 검색을 해보았습니다. 그것에 대한 해결책처럼 보이는 내용이 있었는데 openjdk버전 16이상 부터는 TLS를 사용하여 통신하는 Java 애플리케이션에 대하여 보안적인 이슈로 인한 TLS1.0, TLS1.1의 가용성을 비활성화하여 데이터베이스 접속이 되지 않는 문제라는 것이었습니다. 그렇게 저는 java.security 파일의jdk.tls.disabledAlgorithms에서 TLSv1, TLSv1.1 제거하는 것으로 에러를 해결 할 수 있었습니다.


주로 사용하던 기술과는 달라 처음에는 당황하였지만 동료들과 함께 끊임없이 해결 방안을 찾았습니다. 계속해서 코드를 변경해보면서 결국 문제를 해결할 수 있었습니다. 정확한 답이 없는 상황에서 풀리지 않는 문제에 답답하긴하지만 답을 찾는 그 과정 자체가 즐거웠습니다. 마침내 방법을 찾아내는 짜릿한 경험을 가졌고 함께 성장도 할 수 있었습니다.

웹 어플리케이션 서버 개발을 주로 자바 스프링으로 진행했었습니다. 그런데 점점 다양한 언어, 프레임워크를 사용하여 개발하는 곳이 늘어나고 있는 것을 느꼈습니다. 따라서 저도 다른 언어로 서비스를 만들어보고 싶은 생각이 들었습니다. 튜토리얼부터 차근차근 따라 학습하는 것도 물론 좋지만 서비스를 하나 만들어보는 것이 더 빠르게 배울 수 있을 것 같아 같은 연구실의 연구생들과 서비스를 만들기로 하였습니다.

자바스크립트로 커뮤니티 사이트를 만드는 작은 프로젝트를 시작하였습니다. 저는 백엔드를 맡았고 Express를 사용하여 서버를 구성하였습니다. 백엔드 팀원 5명이 함께 공부하면서 개발을 진행하였습니다. 구현을 함에 있어서 오히려 오류도 많이 나지 않고 자바를 사용할 때보다 더욱 빠르게 서비스를 만들 수 있었습니다. 그래서 게시판, 커뮤니티의 기능을 완성 을 하였고 배포까지 완료했습니다. 그러나 학생들이 모여 만든 사이트였고 네이버 클라우드 플랫폼에 올려 배포하니 작은 서버였지만 매달 나가는 금액이 너무 커 배포를 중단할 수 밖에 없었습니다.


이러한 여러가지 경험을 거치면서, 팀원들과 함께 이야기를 나누며 아이디어를 공유하고 의견을 조율하며 개발하는 것이 개발자로서의 능력을 키워 나갈 수 있는 가장 좋은 방법이라고 느꼈습니다. 다른 사람들과 협력하여 큰 목표를 달성하는 경험을 갖고, 풀리지 않는 문제가 있다면 함께 고민하고 해결해나가고 싶습니다.

# 포트폴리오



## 박준수

백엔드 개발자 박준수입니다.



**Contact.**  
Email. pksjmh5295@gmail.com  
Phone. 010-9123-4567

**Channel.**  
GitHub. <https://github.com/BearWithpy>

## 소개해 드릴게요.

백엔드 서버 개발을 하는 것에 관심이 많습니다.

답이 정해져 있지 않은 문제를 계속해서 만나면서 나의 코드를 쓰는 것을 좋아합니다.

여러 사람과 함께 대화를 많이 나누며 개발하는 것이 즐겁습니다.

## 기술 스택입니다.

- Back-End: Node.js / Express , Java / Spring Boot

너무나도 급박하게 만들었던 나머지...

나중에 피드백 받았었는데,,,

경험이 너무 적은데...?

너에 대해 궁금하지 않아

구체적인 수치를 이용하여 너가  
맡은 파트를 다시 정리 해봐

# 아니... 어째서

ICT 학점연계 프로젝트 인턴십 사업을 통해  
저희 회사에 지원하신 내용보고 메세지 드립니다.  
박준수님 이력 서류를 내부적으로 검토하였는데 서  
류전형에 통과하셨습니다.  
혹시 괜찮으시다면 다음주 월요일이나  
화요일날에 면접을 진행하고싶은데  
가능하실까요? ^^

여기가...?

# 02

## 면접

- 서류 탈탈탈
- 비대면 화상 면접
- 대면 면접



# 서류 탈탈탈!

앞에서 확인하셨지만 면접 한 곳을 붙었습니다.

코딩테스트도 보기도 했지만 오우...

[2022 토스 NEXT] 준수님, 코딩테스트 결과 안내 드립니다.

2022.08.10. 17:04



본 메일은 [2022 토스 NEXT]에 지원하신 분들을 대상으로 발송되는 메일입니다.

준수님, 안녕하세요.  
토스채용팀입니다.

귀한 시간을 내어 코딩테스트에 참여해 주신 점 다시 한번 감사합니다.  
준수님께서 Server 직군으로 제출해주신 테스트 결과물을 확인한 결과, 이번에는 아쉽게도 좋은 소식을 전하지 못하게 되었습니다.

단 몇 개의 문제만으로 준수님이 지닌 가능성을 제대로 볼 수 없었던 것은 토스커뮤니티에게도 큰 아쉬움으로 남습니다.  
어떠한 말로도 아쉬움을 덜어드리지 못하겠지만, 준수님의 부족함이 아닌 토스커뮤니티에 보다 적합한 분을 모시기 위한 결정이라는 점을 너그럽게 이해해주시기를 바랍니다.

준수님이 테스트에 참가하기 위해 투자해 주신 시간은, 토스커뮤니티에게도 큰 혁신을 만들 수 있는 소중한 시간이었습니다.  
그 귀한 시간을 함께해주셨다는 것에 감사드리며, 향후에 더 좋은 기회로 준수님을 다시 뵈 수 있도록 토스커뮤니티 또한 더욱 노력하고 성장하겠습니다.

준수님의 빛나는 앞날을 토스커뮤니티도 늘 함께 응원하겠습니다.

감사합니다.  
토스채용팀 드림

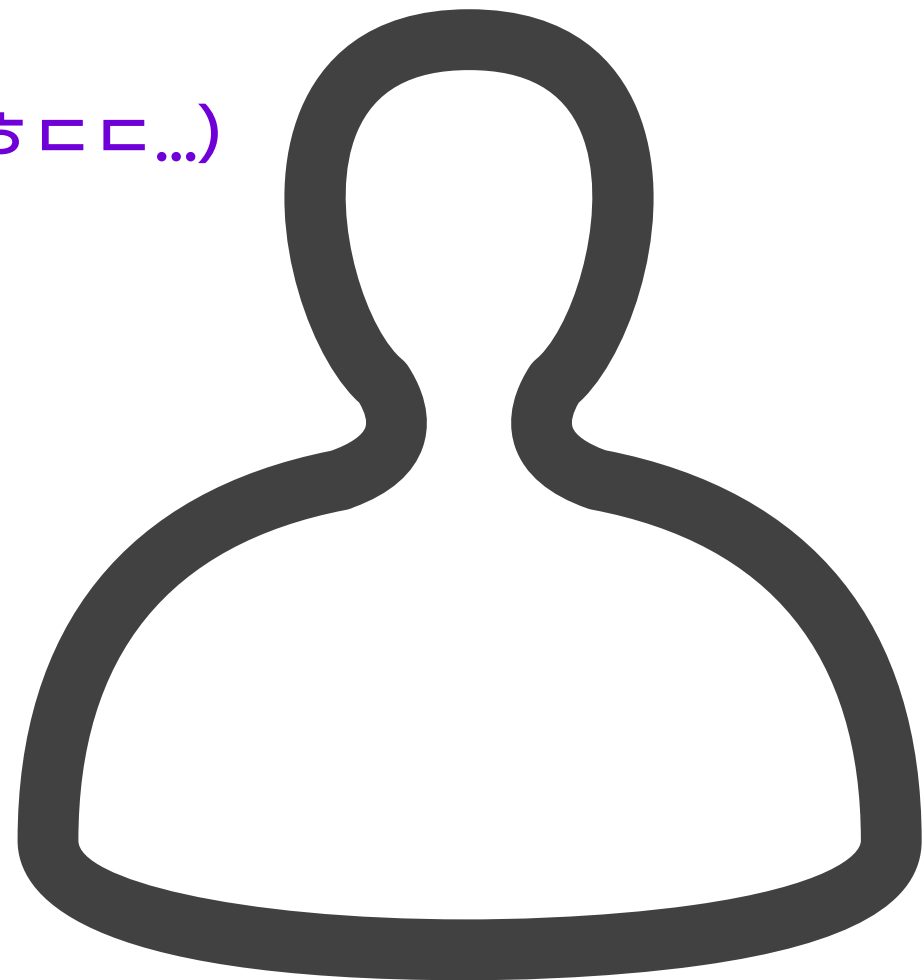
지역	1차 코딩테스트 결과, 합격 소식을 전해드리지 못하게 되었습니다. 지원자 간 형평성을 고려하여 전형 결과 관련 질문에 개별적으로 답변을 드리지 못하는 점 미리 양해 부탁드립니다. 비록 이번에는 인연을 이어가지 못했지만 지원자님의 개발자로서의 성장을 진심으로 응원하겠습니다. 앞날에 행운이 가득하시길 바랍니다.		
서울 서초구	매칭상태	면접일자	기업선택
	서류전형 불합격		
서울 영등포구	서류전형 불합격		

# 비대면 화상면접

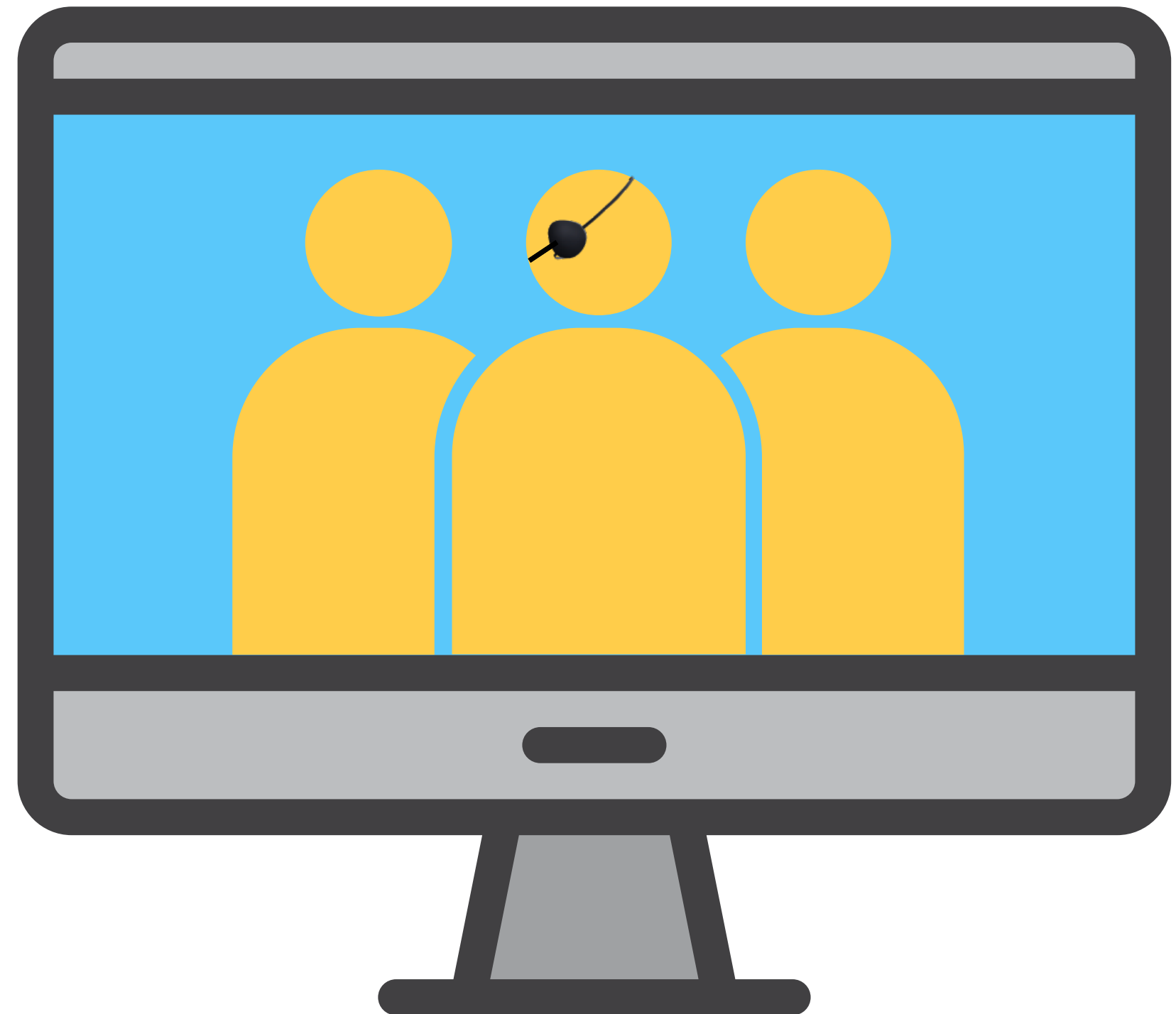
인턴십 이전에 했던 산업 현장 실습

비대면 다대일 면접 (Zoom)

(ㅎㅎ...)



누구인가? 누가 면접 소리를 하였어?



## 비대면 화상면접 (면접관3 : 1)

Q. 자기소개 한 번 부탁드립니다.

Q. (게시판) 프로젝트에서 어떤 파트를 맡아서 개발을 진행했는지 알 수 있을까요?

Q. 혹시 그러면 이 프로젝트에서 사용한 데이터베이스의 종류는 뭔가요?

Q. 그렇다면 NoSQL과 RDB의 차이점이 뭡까요? 그리고 이 프로젝트에서는 왜 이 타입의 데이터베이스를 사용했을까요?

Q. 저희는 자바로 개발을 진행하고 있습니다. 그런데 준수님이 내주신 포트폴리오나 자료에서는 자바의 내용이 많이 없는것 같아서요, 혹시 자바에 대한 설명을 아는대로 전부 말씀해봐 주시겠어요?

Q. 자바의 예외에 대해 질문 드릴게요. Checked Exception과 Unchecked Exception에 대해 말씀해 주실 수 있으신가요?

...이외의 인성 질문은 생략

## 비대면 화상면접

진짜 이 때도 탈탈 털렸습니다.

그래도 답변할 만했던 부분이 많았던 이유?

질문 비율이 프로젝트 관련 경험 및 기술 질문 60%

기타 기술 질문 30%

자기 소개 및 인성 질문 10%

답변을 아예 하지 못한 것은 없었지만.... 결과는 탈락!



## 대면 면접

도착시 문앞에 벨을 누르시거나 제번호로 전화 부탁  
드릴게요~

면접시 따로 챙겨야될 서류는 없고 너무 격식있는 복  
장 보다는 어느정도 편한 복장으로 오시는 걸 추천드  
립니다.

다음주 화요일날 뵈겠습니다~^^

제일 어려운 말. 어느정도 ~

그래서 흰 셔츠와 검은 슬랙스를 입고, 운동화를 신고 갔습니다.



모나미 그 자체

## 대면 면접 (면접관1 : 1)

Q. 신입 분이시니까 그렇게 기술적으로 들어가는 질문은 안 드릴거고, 그냥 대화하듯 편하게 답변해 주시면 될 것 같습니다.

Q. 혹시 자바를 어떤 버전 부터 사용해 보셨을까요?

A. 자바 8버전을 가지고 개발을 처음 진행했습니다.

Q. 그렇다면 자바 8버전에서 새롭게 나온 기능인 Lambda, Functional Interface, Stream, Optional 뭐 이런 것들에 대해 아시는 게 있으실까요?

A. (1차 당황)

Q. Optional 객체에서 orElse()와 orElseGet() 차이점을 설명해주실 수 있을까요?

A. (2차 당황)

# cf. Optional

**optional: null-safe(Null 안전성)한 처리를 위해 사용**

```
Optional<String> optionalValue = Optional.empty();  
String result = optionalValue.orElse("Default Value");  
String res = optionalValue.orElseGet(() -> "Default Value");
```

orElse는 항상 기본 값을 생성하고 반환

orElseGet은 Optional 객체가 비어있을 때만 기본 값을 생성하며, 기본 값이 필요하지 않은 경우에는 불필요한 생성 방지  
따라서 성능에 민감한 상황에서는 orElseGet을 사용하는 것이 유리합니다.



# cf. Functional Interface

functional interface: 하나의 추상 메서드만을 가지고 있는 인터페이스

```
@FunctionalInterface
interface CustomInterface<T> {
    // abstract method 오직 하나
    T myCall();

    // default, static method 는 존재해도 상관없음
    default void printDefault() {
        System.out.println("Hello Default");
    }

    static void printStatic() {
        System.out.println("Hello Static");
    }
}

=====
CustomInterface<String> customInterface = () -> "Hello Custom";

String s = customInterface.myCall();
System.out.println(s);

customInterface.printDefault();
CustomFunctionalInterface.printStatic();
```



# 대면 면접 (면접관1 : 1)

Q. 어느 임의의 값이 담긴 배열이 있다고 가정해 볼게요. 여기에서 중복된 값들을 제거하고 싶는데 어떻게 해야 할까요?

A. 집합을 사용하여 리스트를 한번 걸러낸 후, 리스트를 반환하면 중복된 값이 사라지게 됩니다.

```
import java.util.ArrayList;
import java.util.HashSet;
import java.util.List;

public class Main {
    public static void main(String[] args) {

        List<Integer> numbers = new ArrayList<>();
        numbers.add(1);
        numbers.add(2);
        numbers.add(3);
        numbers.add(2);
        numbers.add(4);
        numbers.add(1);

        HashSet<Integer> uniqueNumbers = new HashSet<>(numbers);
    }
}
```

# 대면 면접 (면접관1 : 1)

Q. 그렇다면 이러한 배열이 있다고 2개 있다고 가정해 볼게요. 또 다기 중복된 값들을 제거하고 2개의 배열을 합치고 싶어요. 어떻게 처리해야 할까요?

```
import java.util.ArrayList;
import java.util.List;
import java.util.stream.Collectors;
import java.util.stream.Stream;

public class main
{
    // tip: arguments are passed via the field below this editor
    public static void main(String[] args)
    {
        List<Integer> list1 = new ArrayList<>();
        list1.add(1); list1.add(2); list1.add(3);
        list1.add(2); list1.add(4); list1.add(1);

        List<Integer> list2 = new ArrayList<>();
        list2.add(3); list2.add(4); list2.add(5);
        list2.add(4); list2.add(4); list2.add(6);

        List<Integer> mergedList = mergeLists(list1, list2);
        System.out.println("Merged List (중복 제거): " + mergedList);
    }

    public static List<Integer> mergeLists(List<Integer> list1,
                                           List<Integer> list2) {
        return Stream.concat(list1.stream(), list2.stream())
            .distinct()
            .collect(Collectors.toList());
    }
}
```

A. (3차 당황;; 내가 말한게 틀린건가...? 왜 다시 묻는 거지? πππ)

Stream.concat 메서드를 사용하여 두 개의 리스트를 합친 다음,  
distinct 메서드를 호출하여 중복값을 제거하고, collect 메서드를 통해  
최종적으로 리스트로 반환합니다.

## 대면 면접 (면접관1 : 1)

Q. 추상클래스와 인터페이스에 대해 설명해주실 수 있으신가요?

A. 추상클래스는 하나 이상의 추상 메서드를 가지는 클래스를 말합니다. 메서드의 선언만 있고, 구현 내용이 없다는 것입니다. new 키워드를 통해 직접 객체를 생성할 수 없다는 특징이 있습니다. 다중 상속을 지원하지 않기 때문에 여러 개의 추상 클래스를 상속할 수는 없습니다.

인터페이스는 메서드의 선언만 있고, 구현 내용이 없다는 것입니다. 일반 메서드 또는 멤버 변수를 구성원으로 가질 수 없습니다. 다중으로 상속이 가능하고 해당 추상메서드들의 구현을 강제합니다.

Q. 그렇다면 인터페이스에는 변수가 존재하지 않나요?

A. 네. (????????????????????????????)

## 대면 면접 (면접관1 : 1)

Q. 쿠키와 세션에 대해 설명해주실 수 있으실까요?

A. 세션은 서버 측에서 상태를 유지하는 방법으로, 클라이언트와 서버 간의 연결을 통해 정보를 저장하고 관리합니다. 웹 브라우저가 서버에 최초로 요청을 보내면, 서버는 해당 클라이언트에 대한 고유한 세션을 생성하고 세션 ID 부여 이후 클라이언트가 서버에 요청을 보낼 때마다, 서버는 세션 ID를 통해 클라이언트를 식별, 해당 세션에 저장된 정보를 활용 세션은 서버 측에 저장되므로, 클라이언트 측에서 직접 접근이 불가능하며 상대적으로 안전한 방식으로 상태를 관리할 수 있습니다. 세션은 일정 시간 또는 세션 유효 기간이 지나면 만료되며, 서버에서 관리되기 때문에 사용자가 브라우저를 종료하거나 웹 페이지를 떠나도 유지될 수 있습니다.

쿠키는 Client 측에서 상태를 유지하는 방법으로, 웹 브라우저에 특정 정보를 저장하여 서버가 해당 정보를 활용할 수 있게 합니다. 서버에서 Client로 HTTP 응답 헤더에 쿠키를 포함하여 클라이언트에게 전송하고, 이후 클라이언트가 동일한 서버에 요청을 보낼 때마다 쿠키가 자동으로 서버로 전송됩니다. 클라이언트는 쿠키를 로컬에 저장하고, 만료 기간을 설정할 수 있으며, 설정된 만료 시간까지 유지됩니다. 쿠키는 클라이언트 측에 저장되기 때문에 보안상의 이유로 민감한 정보는 저장하지 않는 것이 좋습니다. 대신, 세션 ID와 같은 식별자를 저장하여 세션 관리에 활용하거나 사용자 설정 등을 저장하는 데 활용합니다. 클라이언트 측에 저장되기 때문에 브라우저 설정에 따라 쿠키를 차단하거나 삭제가능. 쿠키는 항상 사용 가능한 것은 아닙니다.

## 대면 면접 (면접관1 : 1)

Q. 데이터베이스에서 Optimizer에 대해 설명해주실 수 있으실까요? 그리고 mysql은 어떤 옵티마이저를 사용하나요?

A. (4차 당황;; 진짜 모르겠음;;)

이때부터 어버버, 면접관님이 천천히 생각하면서 차분하게 말씀하셔도 된다고 하시는... 참사 발생

# cf. Optimizer

**Optimizer:** SQL 쿼리의 실행 계획을 생성하는 역할을 수행하는 소프트웨어 컴포넌트, SQL을 가장 빠르고 효율적으로 수행할 최적의 처리경로를 생성해 주는 DBMS 내부의 핵심 엔진

“이 쿼리문을 어떻게 실행시키겠다!”라는 여러가지 실행 계획을 세우고,  
최고의 효율을 갖는 실행계획을 판별한 후 그 실행계획에 따라 쿼리를 수행하게 되는 것입니다.

MySQL은 비용 기반 옵티마이저(Cost-Based Optimizer)를 사용하는  
데이터베이스 관리 시스템입니다.

쿼리 실행 계획을 생성하기 위해 쿼리의 비용과 성능에 관한 통계 정보를 사용합니다.

데이터베이스 통계 정보를 수집하고, 쿼리의 테이블 크기, 인덱스의 선택도 등을 고려하여 실행 계획을 결정합니다.

최적의 실행 계획을 찾기 위해 다양한 실행 계획 후보들의 비용을 비교하여 가장 낮은 비용을 가지는 계획을 선택합니다.

- 규칙 기반 옵티마이저 (Rule-Based Optimizer):
  - 오래된 방식의 옵티마이저로, 사용자가 직접 쿼리 실행 계획을 작성하는 경우 해당 규칙에 따라 실행 계획을 생성합니다.
  - 사용자가 직접 최적의 실행 계획을 작성해야 하기 때문에 쿼리 최적화가 어려웠고, 성능도 미흡합니다.
  - 현재는 거의 사용되지 않고 있으며, 대부분의 DBMS에서는 다른 옵티마이저 유형을 사용합니다.
- 비용 기반 옵티마이저 (Cost-Based Optimizer):
  - 대부분의 현대적인 DBMS에서 사용되는 옵티마이저 유형입니다.
  - 쿼리 실행 계획을 생성하기 위해 쿼리의 비용과 성능에 관한 통계 정보를 사용합니다.
  - 데이터베이스 통계 정보를 수집하고, 쿼리의 테이블 크기, 인덱스의 선택도 등을 고려하여 실행 계획을 결정합니다.
  - 최적의 실행 계획을 찾기 위해 다양한 실행 계획 후보들의 비용을 비교하여 가장 낮은 비용을 가지는 계획을 선택합니다.
- 규칙 기반 옵티마이저와 비용 기반 옵티마이저의 조합:
  - 일부 DBMS에서는 규칙 기반 옵티마이저와 비용 기반 옵티마이저를 조합하여 사용하기도 합니다.
  - 간단한 쿼리는 규칙 기반 옵티마이저로 최적화하고, 복잡한 쿼리는 비용 기반 옵티마이저를 활용하여 최적화하는 방식입니다.
- 통계 기반 옵티마이저 (Statistics-Based Optimizer):
  - 비용 기반 옵티마이저와 유사하지만, 더 정교한 통계 모델을 사용하여 실행 계획을 결정합니다.
  - 통계 모델은 데이터베이스 테이블과 인덱스의 통계 정보를 더욱 정밀하게 수집하여 쿼리 최적화에 활용합니다.
  - 더욱 정교한 쿼리 최적화가 가능하며, 현대적인 대규모 데이터베이스 시스템에 적합합니다.

# 대면 면접

진짜 미친 듯이 탈탈 털렸습니다.

질문 비율이 프로젝트 관련 경험 및 기술 질문 0%  
기타 기술 질문 90%  
자기 소개 및 인성 질문 10%

자바 관련 질문, 알고리즘 질문(?), 데이터베이스 관련 질문

-> 모르면 답할 수 없음;

면접관 입장에서? 그 다음 질문을 이어갈 수 없어서 당혹

알고 있는 부분을 조리있게 말하는 연습, 어느정도 아는 개념은.. 확실하게 알고 있어야 대답할 수 있구나 느낌.



# 03

## 처음부터 다시

- 마지막 학기
- 정처기 / 토익 / 코딩테스트 / 프로젝트



# 결론

- 정보처리기사 실기 준비, 비전공자보다 CS 지식이 부족하다고 느껴지면 안되니까...
- 졸업은 해야지... **토익 목표 800**
- 통곡의 벽, **코딩테스트 공부**하기
- 나는 자바 개발자로 커리어를 시작하고 싶다! 자바 공부 열심히 하자.

# 세상에서 제일 바쁜 4학년 2학기 보내기

- 여러 회사 지원해보기 [서류 - 코딩테스트]

서류 많이 내야 그 중에 몇 개라도 합격도 하고 면접 경험도 쌓아 볼 것 아닌가.

~~이제라도 정신 차리고 해야지... 밥은 먹고 살아야지?~~

- 학교 취업진로상담

# 출처

[1] <https://code-lab1.tistory.com/137>

[2] <https://dev-coco.tistory.com/>

# EOF.

2023. 07. 27 쉬운길 돌아가기 전문가  
KIT SSL Junsu